

NN	NN	TTTTTTTTTT	000000	SSSSSSSS	CCCCCCCC	NN	NN	XX	XX	AAAAAA	BBBBBBBB		
NN	NN	TTTTTTTTTT	000000	SSSSSSSS	CCCCCCCC	NN	NN	XX	XX	AAAAAA	BBBBBBBB		
NN	NN	TT	00	00	SS	CC	NN	NN	XX	XX	AA	BB	
NN	NN	TT	00	00	SS	CC	NN	NN	XX	XX	AA	BB	
NNNN	NN	TT	00	0000	SS	CC	NNNN	NN	XX	XX	AA	BB	
NNNN	NN	TT	00	0000	SS	CC	NNNN	NN	XX	XX	AA	BB	
NN	NN	NN	TT	00	00	SSSSSS	CC	NN	NN	XX	AA	BBBBBBBB	
NN	NN	NN	TT	00	00	SSSSSS	CC	NN	NN	XX	AA	BBBBBBBB	
NN	NNNN	TT	0000	00		SS	CC	NN	NNNN	XX	XX	AAAAAA	BB
NN	NNNN	TT	0000	00		SS	CC	NN	NNNN	XX	XX	AAAAAA	BB
NN	NN	TT	00	00		SS	CC	NN	NN	XX	XX	AA	BB
NN	NN	TT	00	00		SS	CC	NN	NN	XX	XX	AA	BB
NN	NN	TT	000000	SSSSSSSS	CCCCCCCC	NN	NN	XX	XX	AA	BBBBBBBB	
NN	NN	TT	000000	SSSSSSSS	CCCCCCCC	NN	NN	XX	XX	AA	BBBBBBBB	

LL		SSSSSSSS
LL		SSSSSSSS
LL		SS
LL		SS
LL		SS
LL		SSSSSS
LL		SSSSSS
LL		SS
LL		SS
LL		SS
LLLLLLLL		SSSSSSSS
LLLLLLLL		SSSSSSSS

(2)	66	DECLARATIONS
(3)	107	NT\$SCAN_XABCHN - SCAN XAB CHAIN
(4)	396	NT\$SCAN_KEYXAB - SCAN KEY DEFINITION XAB
(4)	397	NT\$SCAN_ALLXAB - SCAN ALLOCATION XAB
(5)	460	NT\$SCAN_NAMBLK - SCAN NAME BLOCK

0000 1 \$BEGIN NTOSCNXAB,000,NFSNETWORK,<SCAN XAB CHAIN>
0000 2
0000 3
0000 4
0000 5 *****
0000 6 *
0000 7 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 8 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 9 * ALL RIGHTS RESERVED.
0000 10 *
0000 11 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 12 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 13 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 14 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 15 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 16 * TRANSFERRED.
0000 17 *
0000 18 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 19 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 20 * CORPORATION.
0000 21 *
0000 22 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 23 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 24 *
0000 25 *
0000 26 *****
0000 27 :
0000 28 :
0000 29 :++
0000 30 : Facility: RMS
0000 31 :
0000 32 : Abstract:
0000 33 :
0000 34 : This module contains routines that scan:
0000 35 : (1) the user XAB chain and examine FAL's capabilities to determine
0000 36 : which DAP Extended Attributes messages should be requested to be
0000 37 : returned by the remote FAL.
0000 38 : (2) the user Name Block and examine FAL's capabilities to determine
0000 39 : if a DAP (resultant) Name message should be requested to be
0000 40 : returned by the remote FAL.
0000 41 :
0000 42 : Environment: VAX/VMS, executive mode
0000 43 :
0000 44 : Author: James A. Krycka, Creation Date: 05-JUN-1979
0000 45 :
0000 46 : Modified By:
0000 47 :
0000 48 : V03-006 JAK0119 J A Krycka 16-JUL-1983
0000 49 : Scan the Journaling XAB and save the JOP field for use by
0000 50 : NTSCREATE.
0000 51 :
0000 52 : V03-005 JAK0115 J A Krycka 29-JUN-1983
0000 53 : Support probe of extended Protection XAB.
0000 54 :
0000 55 : V03-004 KRM0110 K Malik 23-May-1983
0000 56 : Update to support DAP V7.0 spec.
0000 57 :
0000 58 :
0000 59 :
0000 60 :
0000 61 :
0000 62 :
0000 63 :
0000 64 :
0000 65 :
0000 66 :
0000 67 :
0000 68 :
0000 69 :
0000 70 :
0000 71 :
0000 72 :
0000 73 :
0000 74 :
0000 75 :
0000 76 :
0000 77 :
0000 78 :
0000 79 :
0000 80 :
0000 81 :
0000 82 :
0000 83 :
0000 84 :
0000 85 :
0000 86 :
0000 87 :
0000 88 :
0000 89 :
0000 90 :
0000 91 :
0000 92 :
0000 93 :
0000 94 :
0000 95 :
0000 96 :
0000 97 :
0000 98 :
0000 99 :
0000 100 :
0000 101 :
0000 102 :
0000 103 :
0000 104 :
0000 105 :
0000 106 :
0000 107 :
0000 108 :
0000 109 :
0000 110 :
0000 111 :
0000 112 :
0000 113 :
0000 114 :
0000 115 :
0000 116 :
0000 117 :
0000 118 :
0000 119 :
0000 120 :
0000 121 :
0000 122 :
0000 123 :
0000 124 :
0000 125 :
0000 126 :
0000 127 :
0000 128 :
0000 129 :
0000 130 :
0000 131 :
0000 132 :
0000 133 :
0000 134 :
0000 135 :
0000 136 :
0000 137 :
0000 138 :
0000 139 :
0000 140 :
0000 141 :
0000 142 :
0000 143 :
0000 144 :
0000 145 :
0000 146 :
0000 147 :
0000 148 :
0000 149 :
0000 150 :
0000 151 :
0000 152 :
0000 153 :
0000 154 :
0000 155 :
0000 156 :
0000 157 :
0000 158 :
0000 159 :
0000 160 :
0000 161 :
0000 162 :
0000 163 :
0000 164 :
0000 165 :
0000 166 :
0000 167 :
0000 168 :
0000 169 :
0000 170 :
0000 171 :
0000 172 :
0000 173 :
0000 174 :
0000 175 :
0000 176 :
0000 177 :
0000 178 :
0000 179 :
0000 180 :
0000 181 :
0000 182 :
0000 183 :
0000 184 :
0000 185 :
0000 186 :
0000 187 :
0000 188 :
0000 189 :
0000 190 :
0000 191 :
0000 192 :
0000 193 :
0000 194 :
0000 195 :
0000 196 :
0000 197 :
0000 198 :
0000 199 :
0000 200 :
0000 201 :
0000 202 :
0000 203 :
0000 204 :
0000 205 :
0000 206 :
0000 207 :
0000 208 :
0000 209 :
0000 210 :
0000 211 :
0000 212 :
0000 213 :
0000 214 :
0000 215 :
0000 216 :
0000 217 :
0000 218 :
0000 219 :
0000 220 :
0000 221 :
0000 222 :
0000 223 :
0000 224 :
0000 225 :
0000 226 :
0000 227 :
0000 228 :
0000 229 :
0000 230 :
0000 231 :
0000 232 :
0000 233 :
0000 234 :
0000 235 :
0000 236 :
0000 237 :
0000 238 :
0000 239 :
0000 240 :
0000 241 :
0000 242 :
0000 243 :
0000 244 :
0000 245 :
0000 246 :
0000 247 :
0000 248 :
0000 249 :
0000 250 :
0000 251 :
0000 252 :
0000 253 :
0000 254 :
0000 255 :
0000 256 :
0000 257 :
0000 258 :
0000 259 :
0000 260 :
0000 261 :
0000 262 :
0000 263 :
0000 264 :
0000 265 :
0000 266 :
0000 267 :
0000 268 :
0000 269 :
0000 270 :
0000 271 :
0000 272 :
0000 273 :
0000 274 :
0000 275 :
0000 276 :
0000 277 :
0000 278 :
0000 279 :
0000 280 :
0000 281 :
0000 282 :
0000 283 :
0000 284 :
0000 285 :
0000 286 :
0000 287 :
0000 288 :
0000 289 :
0000 290 :
0000 291 :
0000 292 :
0000 293 :
0000 294 :
0000 295 :
0000 296 :
0000 297 :
0000 298 :
0000 299 :
0000 300 :
0000 301 :
0000 302 :
0000 303 :
0000 304 :
0000 305 :
0000 306 :
0000 307 :
0000 308 :
0000 309 :
0000 310 :
0000 311 :
0000 312 :
0000 313 :
0000 314 :
0000 315 :
0000 316 :
0000 317 :
0000 318 :
0000 319 :
0000 320 :
0000 321 :
0000 322 :
0000 323 :
0000 324 :
0000 325 :
0000 326 :
0000 327 :
0000 328 :
0000 329 :
0000 330 :
0000 331 :
0000 332 :
0000 333 :
0000 334 :
0000 335 :
0000 336 :
0000 337 :
0000 338 :
0000 339 :
0000 340 :
0000 341 :
0000 342 :
0000 343 :
0000 344 :
0000 345 :
0000 346 :
0000 347 :
0000 348 :
0000 349 :
0000 350 :
0000 351 :
0000 352 :
0000 353 :
0000 354 :
0000 355 :
0000 356 :
0000 357 :
0000 358 :
0000 359 :
0000 360 :
0000 361 :
0000 362 :
0000 363 :
0000 364 :
0000 365 :
0000 366 :
0000 367 :
0000 368 :
0000 369 :
0000 370 :
0000 371 :
0000 372 :
0000 373 :
0000 374 :
0000 375 :
0000 376 :
0000 377 :
0000 378 :
0000 379 :
0000 380 :
0000 381 :
0000 382 :
0000 383 :
0000 384 :
0000 385 :
0000 386 :
0000 387 :
0000 388 :
0000 389 :
0000 390 :
0000 391 :
0000 392 :
0000 393 :
0000 394 :
0000 395 :
0000 396 :
0000 397 :
0000 398 :
0000 399 :
0000 400 :
0000 401 :
0000 402 :
0000 403 :
0000 404 :
0000 405 :
0000 406 :
0000 407 :
0000 408 :
0000 409 :
0000 410 :
0000 411 :
0000 412 :
0000 413 :
0000 414 :
0000 415 :
0000 416 :
0000 417 :
0000 418 :
0000 419 :
0000 420 :
0000 421 :
0000 422 :
0000 423 :
0000 424 :
0000 425 :
0000 426 :
0000 427 :
0000 428 :
0000 429 :
0000 430 :
0000 431 :
0000 432 :
0000 433 :
0000 434 :
0000 435 :
0000 436 :
0000 437 :
0000 438 :
0000 439 :
0000 440 :
0000 441 :
0000 442 :
0000 443 :
0000 444 :
0000 445 :
0000 446 :
0000 447 :
0000 448 :
0000 449 :
0000 450 :
0000 451 :
0000 452 :
0000 453 :
0000 454 :
0000 455 :
0000 456 :
0000 457 :
0000 458 :
0000 459 :
0000 460 :
0000 461 :
0000 462 :
0000 463 :
0000 464 :
0000 465 :
0000 466 :
0000 467 :
0000 468 :
0000 469 :
0000 470 :
0000 471 :
0000 472 :
0000 473 :
0000 474 :
0000 475 :
0000 476 :
0000 477 :
0000 478 :
0000 479 :
0000 480 :
0000 481 :
0000 482 :
0000 483 :
0000 484 :
0000 485 :
0000 486 :
0000 487 :
0000 488 :
0000 489 :
0000 490 :
0000 491 :
0000 492 :
0000 493 :
0000 494 :
0000 495 :
0000 496 :
0000 497 :
0000 498 :
0000 499 :
0000 500 :
0000 501 :
0000 502 :
0000 503 :
0000 504 :
0000 505 :
0000 506 :
0000 507 :
0000 508 :
0000 509 :
0000 510 :
0000 511 :
0000 512 :
0000 513 :
0000 514 :
0000 515 :
0000 516 :
0000 517 :
0000 518 :
0000 519 :
0000 520 :
0000 521 :
0000 522 :
0000 523 :
0000 524 :
0000 525 :
0000 526 :
0000 527 :
0000 528 :
0000 529 :
0000 530 :
0000 531 :
0000 532 :
0000 533 :
0000 534 :
0000 535 :
0000 536 :
0000 537 :
0000 538 :
0000 539 :
0000 540 :
0000 541 :
0000 542 :
0000 543 :
0000 544 :
0000 545 :
0000 546 :
0000 547 :
0000 548 :
0000 549 :
0000 550 :
0000 551 :
0000 552 :
0000 553 :
0000 554 :
0000 555 :
0000 556 :
0000 557 :
0000 558 :
0000 559 :
0000 560 :
0000 561 :
0000 562 :
0000 563 :
0000 564 :
0000 565 :
0000 566 :
0000 567 :
0000 568 :
0000 569 :
0000 570 :
0000 571 :
0000 572 :
0000 573 :
0000 574 :
0000 575 :
0000 576 :
0000 577 :
0000 578 :
0000 579 :
0000 580 :
0000 581 :
0000 582 :
0000 583 :
0000 584 :
0000 585 :
0000 586 :
0000 587 :
0000 588 :
0000 589 :
0000 590 :
0000 591 :
0000 592 :
0000 593 :
0000 594 :
0000 595 :
0000 596 :
0000 597 :
0000 598 :
0000 5

0000	58	:	V03-003	KPL0001	Peter Lieberwirth	23-May-1983
0000	59	:			Fix branch destinations that are out of range.	
0000	60	:				
0000	61	:	V03-002	KRM0050	K R Malik	02-Jun-1982
0000	62	:			Fix minor bug in NT\$SCAN_NAMBLK.	
0000	63	:				
0000	64	--				

0000 66 .SBTTL DECLARATIONS
0000 67
0000 68 :
0000 69 : Include Files:
0000 70 ;
0000 71
0000 72 \$DAPCNFDEF : Define DAP Configuration message
0000 73 \$DAPACCDEF : Define DAP Access message
0000 74 \$FABDEF : Define File Access Block symbols
0000 75 \$IFBDEF : Define IFAB symbols
0000 76 \$NAMDEF : Define Name Block symbols
0000 77 \$NWADEF : Define Network Work Area symbols
0000 78 \$XABDEF : Define symbols common to all XABs
0000 79 \$XABALLDEF : Define Allocation XAB symbols
0000 80 \$XABDATDEF : Define Date and Time XAB symbols
0000 81 \$XABFHCDDEF : Define File Header Char symbols
0000 82 \$XABKEYDEF : Define Key Definition XAB symbols
0000 83 \$XABPRODEF : Define Protection XAB symbols
0000 84 \$XABRDTDEF : Define Revision Date/Time XAB symbols
0000 85 \$XABSUMDEF : Define Summary XAB symbols
0000 86 : \$XABTRMDEF : Define Terminal XAB symbols
0000 87 : \$XABCXFDEF : Define FAB Context XAB symbols
0000 88 : \$XABCXRDEF : Define RAB Context XAB symbols
0000 89 : \$XABJNLDEF : Define Journal XAB symbols
0000 90
0000 91 :
0000 92 : Macros:
0000 93 :
0000 94 : None
0000 95 :
0000 96 : Equated Symbols:
0000 97 :
0000 98
0000 99 ASSUME NWASQ_FLG EQ 0
0000 100
0000 101 :
0000 102 : Own Storage:
0000 103 :
0000 104 : None
0000 105 :
Ph
--
In
In
Co
Pa
Sy
Pa
Sy
Ps
Cr
As
Th
64
Th
52
34
Ma
--
-S
-S
-S
TO
13
Th
MA

0000 107 .SBTTL NTSSCAN_XABCHN - SCAN XAB CHAIN
0000 108
0000 109 ++
0000 110 NTSSCAN_XABCHN - scans the user XAB chain and examines FAL's capabilities
0000 111 to determine which DAP Attributes and Extended Attributes messages to
0000 112 request the remote FAL to return. It also verifies that all Allocation
0000 113 XABs found are chained sequentially and that all Key Definition XABs
0000 114 found are chained sequentially (i.e., they form sub-chains).
0000 115
0000 116 The message request mask is returned in R2.
0000 117
0000 118 Calling Sequence:
0000 119
0000 120 BSBW NTSSCAN_XABCHN
0000 121
0000 122 Input Parameters:
0000 123
0000 124 R6 Close operation flag
0000 125 R7 NWA (=DAP) address
0000 126 R8 FAB address
0000 127 R9 IFAB address
0000 128 R10 IFAB/FWA address
0000 129 R11 Impure Area address
0000 130
0000 131 Implicit Inputs:
0000 132
0000 133 User ALL, DAT, FHC, JNL, KEY, PRO, RDT, and SUM XABs
0000 134 DAP\$Q_SYS\$CAP bits KEYXAB, ALLXAB, SUMXAB, TIMXAB, PROXAB,
0000 135 CHG\$IM\$CLS, CHG\$PRO\$CLS
0000 136
0000 137 Output Parameters:
0000 138
0000 139 R0 Status code (RMS)
0000 140 R1 Destroyed
0000 141 R2 Message request mask
0000 142 R3-R5 Destroyed
0000 143
0000 144 Implicit Outputs:
0000 145
0000 146 NWASB_ALLXABCNT
0000 147 NWASB_KEYXABCNT
0000 148 NWASB_JNLXABJOP
0000 149 NWASL_ALLXABADR
0000 150 NWASL_DATXABADR
0000 151 NWASL_FHCXABADR
0000 152 NWASL_KEYXABADR
0000 153 NWASL_PROXABADR
0000 154 NWASL_RDTXABADR
0000 155 NWASL_SUMXABADR
0000 156
0000 157 Completion Codes:
0000 158
0000 159 Standard RMS completion codes
0000 160
0000 161 Side Effects:
0000 162
0000 163 User XABs are probed for writeability.

53 24 A8 07	0000 164	:-	
	0000 165	:-	
	0000 166		
	0000 167	NTSSCAN_XABCHN::	Entry point
	0000 168	\$ZERO_FILL-	Zero XAB scan results block
	0000 169	DST=NWAST SCAN(R7)-	Address of block
	0000 170	SIZE=#NWASC_SCAN	Length of block
	000A 171		Zero R2 (request mask) as side effect
	000A 172		of executing a MOVC5 instruction
	000A 173	MOVL FABSL_XAB(R8),R3	Get first XAB address in chain
	000E 174	BRB CHKXAB	;
	0010 175		
	0010 176	:+	
	0010 177	Process next XAB in the chain.	
	0010 178		
	0010 179	Note: XABs not supported by DECnet (if found in the XAB chain) will be	
	0010 180	ignored.	
	0010 181	-	
	0010 182		
	0010 183	ASSUME XAB\$C_DAT EQ 18	
	0010 184	ASSUME XAB\$C_PRO EQ 19	
	0010 185	ASSUME XAB\$C_ALL EQ 20	
	0010 186	ASSUME XAB\$C_KEY EQ 21	
	0010 187	ASSUME XAB\$C_SUM EQ 22	
	0010 188	ASSUME XAB\$C_FHC EQ 29	
	0010 189	ASSUME XAB\$C_RDT EQ 30	
	0010 190	ASSUME XAB\$C_TRM EQ 31	: Not supported for network use
	0010 191	ASSUME XAB\$C_CXF EQ 32	: Not supported for network use
	0010 192	ASSUME XAB\$C_CXR EQ 33	: Not supported for network use
	0010 193	ASSUME XAB\$C_JNL EQ 34	: Not supported for network use
	0010 194		: BUT will be processed to obtain
	0010 195		: JOP field for use by NT\$CREATE
	0010 196		
53 54 53 53	0010 197	NXTXAB: MOVL R3,R4	Save address of current XAB in chain
04 A3 00 30	0013 198	MOVL XAB\$L_NXT(R3),R3	Get address of next XAB in chain
13 13	0017 199	CHKXAB: BEQL EXIT	Branch if none
	0019 200	IFNORD #XAB\$L_NXT+4,(R3),-	Probe for readability thru NXT field
	0019 201	ERRXAB,IFBSB_MODE(R9)	of XAB and branch on failure
ED AF 9F	0020 202	PUSHAB B^NXTXAB	Push return address on stack
	0023 203	\$CASEB SELECTOR=XAB\$B_COD(R3)-	Dispatch to routine to process:
	0023 204	BASE=#XAB\$C_DAT-	
	0023 205	DISPL=<-	
	0023 206	DATE TIME-	XABDAT
	0023 207	PROTECTION-	XABPRO
	0023 208	ALLOCATION-	XABALL
	0023 209	KEY DEFINITION-	XABKEY
	0023 210	SUMMARY-	XABSUM
	0023 211	ERRCOD-	Invalid XAB type
	0023 212	ERRCOD-	Invalid XAB type
	0023 213	ERRCOD-	Invalid XAB type
	0023 214	ERRCOD-	Invalid XAB type
	0023 215	ERRCOD-	Invalid XAB type
	0023 216	ERRCOD-	Invalid XAB type
	0023 217	FILE HEADER-	XABFHC
	0023 218	REV DATE_TIME-	XABRDT
	0023 219	EXIT-	Ignore XABTRM
	0023 220	EXIT-	Ignore XABCXF

```

0023 221          EXIT-          ; Ignore XABCXR
0023 222          JOURNALING-  ; XABJNL
0023 223
0049 224
0049 225 :+
0049 226 : Exit paths.
0049 227 :-
0049 228
0049 229 EXIT: RMSSUC          ; Return success
01 05 004C 230 RSB             ; Exit with RMS code in R0
BA 004D 231 ERRCOD: POPR  #^M<R0> ; Discard return address
004F 232 RMSERR COD
10 11 0054 233 BRB             ; Invalid XAB type code
01 BA 0056 234 ERRIMX: POPR  #^M<R0>
0058 235 RMSERR IMX
07 11 005D 236 BRB             ; Discard return address
03 BA 005F 237 ERRXAB2: POPR  #^M<R0,R1>
0061 238 ERRXAB: RMSERR XAB  ; Duplicate XAB or XABs are not dense
0062 239 SETSTV: MOVL  R3,FAB$L_STV(R8) ; Discard return addresses
006A 240 RSB             ; XAB too short or not accessible
006B 241
006B 242 :+
006B 243 : This routine checks the control block for correct length and writeability.
006B 244 :-
006B 245
006B 246 VALIDATE_XAB:          ; Entry point
006B 247 MOVZBL XAB$B_BLN(R3),R0 ; Get stated length of block
50 01 A3 9A 006B 248 CMPL R0,R1 ; Compare against expected length
51 50 D1 006F 249 BLSSU ERRXAB2 ; Branch if too small
EB 1F 0072 250 IFNOWRT R0,(R3),ERRXAB2,- ; Probe for writeability and branch on
0074 251 IFBS$B_MODE(R9)        ; failure
05 007B 252 RSB             ; Exit
007C 253
007C 254 :+
007C 255 : This routine handles the Date and Time XAB.
007C 256 :-
007C 257
007C 258 DATE_TIME:          ; Entry point
51 24 9A 007C 259 MOVZBL #XAB$C_DATLEN_V2,R1 ; Get minimum (i.e., V2) length of XAB
EA 10 007F 260 BSSB  VALIDATE_XAB ; Check length and accessibility
0104 C7 D5 0081 261 TSTL  NWASL_DATXABADR(R7) ; Declare error as this is a duplicate
CF 12 0085 262 BNEQ  ERRIMX ; XAB
0104 C7 53 D0 0087 263 MOVL  R3,NWASL_DATXABADR(R7) ; Save address of Date and Time XAB
1A E1 008C 264 BBC   #DAP$V_TIMXAB,- ; Branch if Date and Time message is
07 28 A7 008E 265 DAP$Q SYSCAP(R7),10$ ; not supported by partner
04 56 E8 0091 266 BLBS  R6,10$ ; This XAB is not an input on close
05 0094 267 SSETBIT #DAP$V_DSP_TIM,R2 ; Update request mask
0099 268 10$: RSB             ; Exit
0099 269
0099 270 :+
0099 271 : This routine handles the Protection XAB.
0099 272 :-
0099 273
0099 274 PROTECTION:          ; Entry point
51 10 9A 0099 275 MOVZBL #XAB$C_PROLEN_V3,R1 ; Get minimum (i.e., V3) length of XAB
CD 10 009C 276 BSSB  VALIDATE_XAB ; Check length and accessibility
0110 C7 D5 009E 277 TSTL  NWASL_PR0XABADR(R7) ; Declare error as this is a duplicate

```

0110 C7 B2 12 00A2 278 BNEQ ERRIMX ; XAB
55 D0 00A4 279 MOVL R3,NWASL PROXABADR(R7) Save address of Protection XAB
1B E1 00A9 280 BBC #DAPSV PROXAB,- Branch if Protection message is
OC 28 A7 00AB 281 DAPSQ SYSCAP(R7),20\$ not supported by partner
05 56 E9 00AE 282 BLBC R6,10\$ An additional system capabilities
2C E1 00B1 283 BBC #DAPSV CHGPROCLS,- check is required if this is a
04 28 A7 00B3 284 DAPSQ SYSCAP(R7),20\$ change operation
00B6 285 10\$: SSETBIT #DAPSV_DSP_PRO,R2 Update request mask
05 00BA 286 20\$: RSB ; Exit
00BB 287
00BB 288 ;+
00BB 289 ; This routine handles the Allocation XAB.
00BB 290 ;-
00BB 291
00BB 292 ALLOCATION: ; Entry point
51 20 9A 00BB 293 MOVZBL #XABSC_ALLLEN,R1 Get length of XAB
AB 10 00BE 294 BSBW VALIDATE XAB Check length and accessibility
01 011C C7 96 00C0 295 INCB NWASB_ALLXABCNT(R7) Increment XAB counter
011C C7 91 00C4 296 CMPB NWASB_ALLXABCNT(R7),#1 Branch if this is first
08 13 00C9 297 BEQL 10\$ Allocation XAB in chain
14 64 91 00CB 298 CMPB XABSB_COD(R4),#XABSC_ALL Check previous XAB in chain;
11 13 00CE 299 BEQL 20\$ it must also be an Allocation XAB
FF83 31 00D0 300 BRW ERRIMX else this XAB is out of order
0100 C7 53 D0 00D3 301 10\$: MOVL R3,NWASL ALLXABADR(R7) Save address of first Allocation XAB
17 E1 00D8 302 BBC #DAPSV_ALLXAB,- Branch if Allocation message is
04 28 A7 00DA 303 DAPSQ SYSCAP(R7),20\$ not supported by partner
05 00D9 304 SSETBIT #DAPSV_DSP_ALL,R2 Update request mask
00E1 305 20\$: RSB ; Exit
00E2 306
00E2 307 ;+
00E2 308 ; This routine handles the Key Definition XAB.
00E2 309 ;-
00E2 310
00E2 311 KEY_DEFINITION: ; Entry point
51 40 8F 9A 00E2 312 MOVZBL #XABSC_KEYLEN_V2,R1 Get minimum (i.e., V2) length of XAB
FF82 30 00E6 313 BSBW VALIDATE XAB Check length and accessibility
01 011D C7 96 00E9 314 INCB NWASB_KEYXABCNT(R7) Increment XAB counter
011D C7 91 00ED 315 CMPB NWASB_KEYXABCNT(R7),#1 Branch if this is first
08 13 00F2 316 BEQL 10\$ Key Definition XAB in chain
15 64 91 00F4 317 CMPB XABSB_COD(R4),#XABSC_KEY Check previous XAB in chain;
11 13 00F7 318 BEQL 20\$ it must also be a Key Definition XAB
FF5A 31 00F9 319 BRW ERRIMX else this XAB is out of order
010C C7 53 D0 00FC 320 10\$: MOVL R3,NWASL KEYXABADR(R7) Save address of first Key Def XAB
16 E1 0101 321 BBC #DAPSV_KEYXAB,- Branch if Key Definition message is
04 28 A7 0103 322 DAPSQ SYSCAP(R7),20\$ not supported by partner
05 0106 323 SSETBIT #DAPSV_DSP_KEY,R2 Update request mask
010A 324 20\$: RSB ; Exit
010B 325
010B 326 ;+
010B 327 ; This routine handles the Summary XAB.
010B 328 ;-
010B 329
010B 330 SUMMARY: ; Entry point
51 OC 9A 010B 331 MOVZBL #XABSC_SUMLEN,R1 Get length of XAB
FF5A 30 010E 332 BSBW VALIDATE XAB Check length and accessibility
0118 C7 D5 0111 333 TSTL NWASL SUMXABADR(R7) Declare error as this is a duplicate
55 12 0115 334 BNEQ ERRIMX1 XAB

SCAN XAB CHAIN
NT\$SCAN_XABCHN - SCAN XAB CHAIN

0118 C7 53 00 0117 335 MOVL R3,NWASL SUMXABADR(R7) ; Save address of Summary XAB
18 E1 011C 336 BBC #DAPSV SOMXAB,-
04 28 A7 011E 337 DAP\$Q SYSCAP(R7),10\$ Branch if Summary message is
05 0125 338 S\$ETBIT #DAP\$D_DSP_SUM,R2 not supported by partner
0126 339 10\$: RSB Update request mask
0126 340 : Exit
0126 341 :+
0126 342 : This routine handles the File Header Characteristics XAB.
0126 343 : Note: The File Header Characteristics XAB is supported in DAP through the
0126 344 : DAP Attributes message. Thus there is no system capabilities check
0126 345 : associated with it.
0126 346 :
0126 347 :
0126 348 :
0126 349 FILE_HEADER: 350 MOVZBL #XABSC FHCLEN,R1 ; Entry point
51 2C 9A 0126 351 BSBW VALIDATE XAB ; Get length of XAB
FF 3F 30 0129 352 TSTL NWASL FH\$XABADR(R7) ; Check length and accessibility
0108 C7 D5 012C 353 BNEQ ERRIMX1 ; Declare error as this is a duplicate
3A 12 0130 354 MOVL R3,NWASL FH\$XABADR(R7) XAB
0108 C7 53 00 0132 355 S\$ETBIT #DAP\$V_DSP_ATT,R2 ; Save address of File Header Char XAB
05 0137 356 RSB ; Update request mask
013C 357 : Exit
013C 358 :+
013C 359 : This routine handles the Revision Date and Time XAB.
013C 360 :
013C 361 : Note: Both the Date and Time XAB and the Revision Date and Time XAB are
013C 362 : supported in DAP through the DAP Date and Time message.
013C 363 :
013C 364 :
013C 365 REV_DATE_TIME: 366 MOVZBL #XABSC RDTLEN,R1 ; Entry point
51 14 9A 013C 367 BSBW VALIDATE XAB ; Get length of XAB
FF 29 30 013F 368 TSTL NWASL RDTXABADR(R7) ; Check length and accessibility
0114 C7 D5 0142 369 BNEQ ERRIMX1 ; Declare error as this is a duplicate
24 12 0146 370 MOVL R3,NWASL RDTXABADR(R7) XAB
0114 C7 53 00 0148 371 BBC #DAP\$V TIMXAB,-
1A E1 014D 372 DAP\$Q SYSCAP(R7),20\$ Branch if Date and Time message
0C 28 A7 014F 373 BLBC R6.10\$; not supported by partner
05 56 E9 0152 374 BBC #DAP\$V CHGTIMCLS,-
28 E1 0155 375 DAP\$Q SYSCAP(R7),20\$ An additional system capabilities
04 28 A7 0157 376 10\$: S\$ETBIT #DAP\$D_DSP_TIM,R2 check is required if this is a
05 015E 377 20\$: RSB change operation
015F 378 : Update request mask
015F 379 : Exit
015F 380 :+
015F 381 : This routine handles the Journaling XAB.
015F 382 : Note: This XAB is not supported for network use--it will be ignored unless
015F 383 : the journaling options field is non-zero on create. Consequently, this
015F 384 : routine saves the JOP field in the NWA for use by NT\$CREATE.
015F 385 :
015F 386 :
015F 387 JOURNALING: 388 MOVZBL #XABSC JNLLEN,R1 ; Entry point
51 3C 9A 015F 389 BSBW VALIDATE XAB ; Get length of XAB
FF 06 30 0162 390 MOVW XABSW_JOP(R3),- ; Check length and accessibility
08 A3 B0 0165 391 NWASW_JNLXABJOP(R7) ; Save journaling options value in NWA

NTOSCNXAB
V04-000

SCAN XAB CHAIN
NT\$SCAN_XABCHN - SCAN XAB CHAIN

M 6

16-SEP-1984 00:07:06 VAX/VMS Macro V04-00
5-SEP-1984 16:21:07 [RMS.SRC]NTOSCNXAB.MAR;1

Page 9
(3)

N
VI

05	016B	392	RSB	: Exit	
016C	393				
FEE7	31	016C	394 ERRIMX1:BRW	ERRIMX	: Branch aid

016F 396 .SBTTL NTSSCAN_KEYXAB - SCAN KEY DEFINITION XAB
 016F 397 .SBTTL NTSSCAN_ALLXAB - SCAN ALLOCATION XAB
 016F 398 ++
 016F 399 NTSSCAN_KEYXAB - scans a specific Key Definition XAB without scanning the
 016F 400 entire XAB chain.
 016F 401 NTSSCAN_ALLXAB - scans a specific Allocation XAB without scanning the
 016F 402 entire XAB chain.
 016F 403
 016F 404 Calling Sequence:
 016F 405
 016F 406 BSBW NTSSCAN_KEYXAB
 016F 407 BSBW NTSSCAN_ALLXAB
 016F 408
 016F 409 Input Parameters:
 016F 410
 016F 411 R6 Allocation or Key Definition XAB address
 016F 412 R7 NWA (=DAP) address
 016F 413 R8 FAB address
 016F 414 R9 IFAB address
 016F 415 R10 IFAB/FWA address
 016F 416 R11 Impure Area address
 016F 417
 016F 418 Implicit Inputs:
 016F 419
 016F 420 None
 016F 421
 016F 422 Output Parameters:
 016F 423
 016F 424 R0 Status code (RMS)
 016F 425 R1 Destroyed
 016F 426 R3 Destroyed
 016F 427
 016F 428 Implicit Outputs:
 016F 429
 016F 430 None
 016F 431
 016F 432 Completion Codes:
 016F 433 Standard RMS Completion codes
 016F 434
 016F 435
 016F 436 Side Effects:
 016F 437
 016F 438 User XAB is probed for writeability
 016F 439
 016F 440 --
 016F 441
 016F 442 NTSSCAN_KEYXAB:: : Entry point
 016F 443 MOVZBL #XABSC_KEYLEN_V2,R1 : Get minimum (i.e., V2) length of XAB
 016F 444 BRB COMMON_SCAN : Join common code
 016F 445 NTSSCAN_ALLXAB:: : Entry point
 016F 446 MOVZBL #XABSC_ALLLEN,R1 : Get length of XAB
 016F 447 COMMON_SCAN: : Common code
 016F 448 MOVL R6,R3 : Get address of XAB to probe
 016F 449 IFNORD #XABSL_NXT+4,(R3) - : Probe for readability thru NXT field
 016F 450 10\$ IFBSB MODE(R9) - of XAB and branch on failure
 016F 451 MOVZBL XAB\$B_BLN(R3),R0 : Get stated length of block
 016F 452 CMPL R0,R1 : Compare against expected length

51 40 8F 9A 016F 443 MOVZBL #XABSC_KEYLEN_V2,R1 : Entry point
 03 11 0173 444 BRB COMMON_SCAN : Get minimum (i.e., V2) length of XAB
 51 20 9A 0175 445 NTSSCAN_ALLXAB:: : Join common code
 0175 446 MOVZBL #XABSC_ALLLEN,R1 : Entry point
 53 56 D0 0178 447 COMMON_SCAN: : Get length of XAB
 0178 448 MOVL R6,R3 : Common code
 0178 449 IFNORD #XABSL_NXT+4,(R3) - : Get address of XAB to probe
 0178 450 10\$ IFBSB MODE(R9) - : Probe for readability thru NXT field
 50 01 A3 9A 0182 451 MOVZBL XAB\$B_BLN(R3),R0 : of XAB and branch on failure
 51 50 D1 0186 452 CMPL R0,R1 : Get stated length of block
 : Compare against expected length

OB	1F	0189	453	BLSSU	10\$		
		0188	454	IFNOWRT	RO (R3)	10\$ -	: Branch if too small
		0188	455		IFBSB	_MODE(R9)	: Probe for writeability and branch on
		0192	456	RMSSUC			: failure
	05	0195	457	RSB			: Return success
FEC8	31	0196	458	BRW	ERRXAB		: Exit
							: Failure

0199 460 .SBTTL NTSSCAN_NAMBLK - SCAN NAME BLOCK

0199 461

0199 462 ++

0199 463 NTSSCAN_NAMBLK - scans the user Name Block and checks FAL's capabilities

0199 464 to determine if a DAP (resultant) Name message should be requested

0199 465 to be returned by the remote FAL.

0199 466

0199 467 An updated message request mask is returned in R2.

0199 468

0199 469 Calling Sequence:

0199 470 BSBW NTSSCAN_NAMBLK

0199 471

0199 472

0199 473 Input Parameters:

0199 474

0199 475 R2 Message request mask

0199 476 R7 NWA (=DAP) address

0199 477 R8 FAB address

0199 478 R9 IFAB address

0199 479 R10 IFAB/FWA address

0199 480 R11 Impure Area address

0199 481

0199 482 Implicit Inputs:

0199 483 User Name Block

0199 484

0199 485

0199 486 Output Parameters:

0199 487

0199 488 R0 Status code (RMS)

0199 489 R1 Destroyed

0199 490 R2 Updated message request mask

0199 491

0199 492 Implicit Outputs:

0199 493 None

0199 494

0199 495

0199 496 Completion Codes:

0199 497 Standard RMS Completion codes

0199 498

0199 499

0199 500 Side Effects:

0199 501 User Name Block is probed for writeability

0199 502

0199 503

0199 504

0199 505

0199 506 NTSSCAN_NAMBLK::

57 00C0 8F BB 0199 507 PUSHR #^M<R6,R7> : Entry point

28 A8 D0 019D 508 MOVL FABSL_NAM(R8),R7 : Save registers used

19 13 01A1 509 BEQL 10\$: Get Name block address

FESA 30 01A3 510 BSBW RMSCHKNAM : Branch if none

13 50 E9 01A6 511 BLBC R0,10\$: Check Name block validity

04 A7 D5 01A9 512 TSTL NAMSL_RSA(R7) : Branch on error

0E 13 01AC 513 BEQL 10\$: Check for resultant string address

00C0 8F 8A 01AE 514 POPR #^M<R6,R7> : Branch if none

28 E1 01B2 515 BBC #DAPSV_NAMMSG,- : Restore registers

09 28 A7 01B4 516 DAPSQ_SYSACP(R7),20\$: Branch if partner does not support

NT
VO

00C0 8F 01B7 517 \$SETBIT #DAP\$V_DSP_NAM,R2 ; Request Name message
05 01B8 518 RSB ; Exit
BA 01BC 519 10\$: POPR #^M<R6,R7> ; Restore registers
05 01C0 520 20\$: RSB ; Exit
01C1 521
01C1 522 .END ; End of module

SS_PSECT_EP	= 00000000	IFBSB_MODE	= 0000006A
SSCOUNT	= 00000011	JOURNALING	0000315F R 01
SSRMSTEST	= 0000001A	KEY DEFINITION	000000E2 R 01
SSRMS_PBUGCHK	= 00000010	NAM\$L RSA	= 00000004
SSRMS_TBUGCHK	= 00000008	NT\$SCAN_ALLXAB	00000175 RG 01
SSRMS_UMODE	= 00000004	NT\$SCAN_KEYXAB	0000016F RG 01
ALLOCATION	00000088 R 01	NT\$SCAN_NAMBLK	00000199 RG 01
CHKXAB	00000017 R 01	NT\$SCAN_XABCHN	00000000 RG 01
COMMON_SCAN	00000178 R 01	NW\$B_ALLXABCNT	0000011C
DAPSB_ACCFUNC	00000040	NW\$B_DAP_RAC	000000C9
DAPSB_ACCEPT	00000041	NW\$B_FILESYS	000000C5
DAPSB_DECVER	00000047	NW\$B_KEYXABCNT	0000011D
DAPSB_ECONUM	00000045	NW\$B_NETSTRSIZ	0000016F
DAPSB_FAC	00000042	NW\$B_NODBUFSIZ	00000168
DAPSB_FILESYS	00000043	NW\$B_ORG	000000C6
DAPSB_OSTYPE	00000042	NW\$B_OSTYPE	000000C4
DAPSB_SHR	00000043	NW\$B_RFM	000000C7
DAPSB_USPNUM	00000046	NW\$B_RMS_RAC	000000C8
DAPSB_USRVER	00000048	NW\$C_BLN	00000800
DAPSB_VERNUM	00000044	NW\$C_SCAN	= 0000020
DAPSM_DSP_3NAM	= 00000200	NW\$K_BLN	00000800
DAPSM_GET	= 00000002	NW\$L_ALLXABADR	00000100
DAPSM_GO_NOGO	= 00000010	NW\$L_DATXABADR	00000104
DAPSM_MSE	= 00000010	NW\$L_DEV	000000C0
DAPSM_TMP1\$	= 000000C0	NW\$L_FHCXABADR	00000108
DAPSM_TMP2\$	= 0000FC00	NW\$L_KEYXABADR	0000010C
DAPSQ_FILESPEC	00000044	NW\$L_MSG_MASK	000000D4
DAPSQ_PASSWORD	00000050	NW\$L_PR0XABADR	00000110
DAPSQ_SYSCAP	00000028	NW\$L_RDTXABADR	00000114
DAPSV_ALLXAB	= 00000017	NW\$L_SAVE_FLGS	00000128
DAPSV_CHGPROCLS	= 0000002C	NW\$L_SUMXABADR	00000118
DAPSV_CHGTIMCLS	= 0000002B	NW\$L_THREAD	000000FC
DAPSV_DSP_ALL	= 00000002	NW\$L_XLTATTR	00000238
DAPSV_DSP_ATT	= 00000000	NW\$L_XLTBUFLG	0000022C
DAPSV_DSP_KEY	= 00000001	NW\$L_XLTCNT	00000228
DAPSV_DSP_NAM	= 00000008	NW\$L_XLTMAXIDX	00000234
DAPSV_DSP_PRO	= 00000005	NW\$L_XLTSIZ	00000230
DAPSV_DSP_SUM	= 00000003	NW\$Q_ACS	00000244
DAPSV_DSP_TIM	= 00000004	NW\$Q_BIGBUF	00000170
DAPSV_KEYXAB	= 00000016	NW\$Q_BLD	000000F0
DAPSV_NAMMSG	= 00000028	NW\$Q_FLG	00000000
DAPSV_PROXAB	= 0000001B	NW\$Q_INODE	0000025C
DAPSV_SUMXAB	= 00000018	NW\$Q_IOSB	000000D8
DAPSV_TIMXAB	= 0000001A	NW\$Q_LNODE	00000160
DAPSW_BUFSIZ	= 00000040	NW\$Q_LOGNAME	0000023C
DAPSW_DISPLAY1	= 0000004C	NW\$Q_NCB	00000264
DATE_TIME	0000007C R 01	NW\$Q_RCV	000000E0
ERRC0D	0000004D R 01	NW\$Q_SAVE_DESC	00000120
ERRIMX	00000056 R 01	NW\$Q_XLTBUF1	0000024C
ERRIMX1	00000016C R 01	NW\$Q_XLTBUF2	00000254
ERRXAB	00000061 R 01	NW\$Q_XMT	000000E8
ERRXAB2	0000005F R 01	NW\$T_ACSBUF	0000026C
EXIT	00000049 R 01	NW\$T_AUXBUF	000005E0
FABSL_NAM	= 00000028	NW\$T_DAP	00000000
FABSL_STV	= 0000000C	NW\$T_INODEBUF	000004AC
FABSL_XAB	= 00000024	NW\$T_ITM_ATTR	00000200
FILE_HEADER	00000126 R 01	NW\$T_ITM_END	00000224

NWAST_ITM_LST	00000200
NWAST_ITM_MAXINDEX	00000218
NWAST_ITM_STRING	0000020C
NWAST_NCBBUF	0000052C
NWAST_NODEBUF	00000169
NWAST_RCVBUF	000001A0
NWAST_SCAN	00000100
NWAST_TEMP	00000120
NWAST_XLTBUF1	000002AC
NWAST_XLTBUF2	000003AC
NWAST_XMTBUF	000003C0
NWASW_BUILD	000000D2
NWASW_DAPBUFSIZ	000000CA
NWASW_DIR_OFF	000000CC
NWASW_DISPLAY	000000D0
NWASW_FIL_OFF	000000CE
NWASW_JNLXABJOP	0000011E
NXTXAB	00000010 R 01
PROTECTION	00000099 R 01
REV_DATE_TIME	0000013C R 01
RMSSCHKNAME	***** X 01
RMSS_COD	***** X 01
RMSS_IMX	***** X 01
RMSS_XAB	***** X 01
SETSTV	00000066 R 01
SUMMARY	0000010B R 01
VALIDATE_XAB	0000006B R 01
XAB\$B_BLN	= 00000001
XAB\$B_COD	= 00000000
XAB\$C_ALL	= 00000014
XAB\$C_ALLEN	= 00000020
XAB\$C_DAT	= 00000012
XAB\$C_DATLEN_V2	= 00000024
XAB\$C_FHC	= 0000001D
XAB\$C_FHCLEN	= 0000002C
XAB\$C_JNL	= 00000022
XAB\$C_JNLLEN	= 0000003C
XAB\$C_KEY	= 00000015
XAB\$C_KEYLEN_V2	= 00000040
XAB\$C_PRO	= 00000013
XAB\$C_PROLEN_V3	= 00000010
XAB\$C_RDT	= 0000001E
XAB\$C_RDTLEN	= 00000014
XAB\$C_SUM	= 00000016
XAB\$C_SUMLEN	= 0000000C
XAB\$L_NXT	= 00000004
XAB\$W_JOP	= 00000008

! Psect synopsis !

PSECT name

ABS
NET\$NETWORK
SABS\$

Allocation	PSECT No.	Attributes
-----	-----	-----
000000000 (0.) 00 (0.) NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE		
000001C1 (449.) 01 (1.) PIC USR CON REL GBL NOSHR EXE RD NOWRT NOVEC BYTE		
00000800 (2048.) 02 (2.) NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE		

+-----+
! Performance indicators !
+-----+

Phase	Page faults	CPU Time	Elapsed Time
Initialization	32	00:00:00.08	00:00:01.05
Command processing	143	00:00:00.80	00:00:04.52
Pass 1	334	00:00:12.16	00:00:35.27
Symbol table sort	0	00:00:01.49	00:00:03.11
Pass 2	103	00:00:02.39	00:00:09.14
Symbol table output	20	00:00:00.15	00:00:00.38
Psect synopsis output	1	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	635	00:00:17.10	00:00:53.51

The working set limit was 1500 pages.

64416 bytes (126 pages) of virtual memory were used to buffer the intermediate code.

There were 60 pages of symbol table space allocated to hold 1101 non-local and 22 local symbols.

522 source lines were read in Pass 1, producing 14 object records in Pass 2.

34 pages of virtual memory were used to define 33 macros.

+-----+
! Macro library statistics !
+-----+

Macro library name	Macros defined
\$255\$DUA28:[RMS.OBJ]RMS.MLB;1	23
\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	2
\$255\$DUA28:[SYSLIB]STARLET.MLB;2	4
TOTALS (all libraries)	29

1326 GETS were required to define 29 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LI\$S:NTOSCNXAB/0BJ=0BJ\$:NTOSCNXAB MSRC\$:NTOSCNXAB/UPDATE=(ENH\$:NTOSCNXAB)+EXECMLS/LIB+LIB\$:RMS/LIB

0317 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY